Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	8052
Core Size	8-Bit
Speed	40MHz
Connectivity	EBI/EMI, UART/USART
Peripherals	POR, WDT
Number of I/O	32
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/nuvoton-technology-corporation-america/w78e058b40dl

8. TIMING WAVEFORMS 25
8.1 Program Fetch Cycle 25
8.2 Data Read Cycle 25
8.3 Data Write Cycle 26
8.4 Port Access Cycle 26
9. TYPICAL APPLICATION CIRCUITS 27
9.1 Expanded External Program Memory and Crystal 27
9.2 Expanded External Data Memory and Oscillator 28
10. PACKAGE DIMENSIONS 29
10.1 40-pin DIP 29
10.2 44-pin PLCC 29
10.3 44-pin PQFP 30
11. APPLICATION NOTES 31
11.1 In-system Programming Software Examples 31
12. REVISION HISTORY 36

```
MAIN_4K:
        MOV SP, #C0H
        MOV CHPENR, #87H   ; CHPENR = 87H, CHPCON WRITE ENABLE.
        MOV CHPENR, #59H   ; CHPENR = 59H, CHPCON WRITE ENABLE.
        MOV A, CHPCON
        ANL A, #80H
        CJNE A, #80H, UPDATE_32K   ; CHECK  F04KBOOT MODE ?

        MOV CHPCON, #03H  ; CHPCON = 03H, ENABLE IN-SYSTEM PROGRAMMING.
        MOV CHPENR, #00H  ; DISABLE CHPCON WRITE ATTRIBUTE

        MOV TCON, #00H      ; TCON = 00H, TR = 0 TIMER0 STOP
        MOV TMOD, #01H      ; TMOD = 01H, SET TIMER0 A 16BIT TIMER
        MOV IP, #00H        ; IP = 00H
        MOV IE, #82H        ; IE = 82H, TIMER0 INTERRUPT ENABLED
        MOV R6, #F0H
        MOV R7, #FFH
        MOV TL0, R6
        MOV TH0, R7
        MOV TCON, #10H      ; TCON = 10H, TR0 = 1, GO
        MOV PCON, #01H      ; ENTER IDLE MODE

UPDATE_32K:
        MOV CHPENR, #00H  ; DISABLE CHPCON WRITE-ATTRIBUTE
        MOV TCON, #00H      ; TCON = 00H , TR = 0 TIM0 STOP
        MOV IP, #00H        ; IP = 00H
        MOV IE, #82H        ; IE = 82H, TIMER0 INTERRUPT ENABLED
        MOV TMOD, #01H      ; TMOD = 01H, MODE1
        MOV R6, #E0H        ; SET WAKE-UP TIME FOR ERASE OPERATION, ABOUT 15 mS. DEPENDING
                            ; ON USER'S SYSTEM CLOCK RATE.
        MOV R7, #B1H
        MOV TL0, R6
        MOV TH0, R7

ERASE_P_4K:
        MOV SFRCN, #22H     ; SFRCN(C7H) = 22H  ERASE 32K
        MOV TCON, #10H      ; TCON = 10H, TR0 = 1,GO
        MOV PCON, #01H      ; ENTER IDLE MODE (FOR ERASE OPERATION)

;*********************************************************************
;* BLANK CHECK
;*********************************************************************
        MOV SFRCN, #0H      ; READ 32KB APROM MODE
        MOV SFRAH, #0H      ; START ADDRESS = 0H
        MOV SFRAL, #0H
        MOV R6, #FEH        ; SET TIMER FOR READ OPERATION, ABOUT 1.5 µS.
        MOV R7, #FFH
        MOV TL0, R6
        MOV TH0, R7

BLANK_CHECK_LOOP:
        SETB TR0            ; ENABLE TIMER 0
        MOV PCON, #01H      ; ENTER IDLE MODE
        MOV A, SFRFD        ; READ ONE BYTE
        CJNE A, #FFH, BLANK_CHECK_ERROR
```

```
READ_VERIFY_32K:
        MOV SFRAL, R2           ; SFRAL(C4H) = LOW ADDRESS
        MOV TCON, #10H          ; TCON = 10H, TR0 = 1,GO
        MOV PCON, #01H
        INC R2
        MOVX A, @DPTR
        INC DPTR
        CJNE A, SFRFD, ERROR_32K
        CJNE R2, #0H, READ_VERIFY_32K
        INC R1
        MOV SFRAH, R1
        CJNE R1, #80H, READ_VERIFY_32K

;*****************************************************************************
;* PROGRAMMING COMPLETLY, SOFTWARE RESET CPU
;*****************************************************************************
        MOV CHPENR, #87H        ; CHPENR = 87H
        MOV CHPENR, #59H        ; CHPENR = 59H
        MOV CHPCON, #83H        ; CHPCON = 83H, SOFTWARE RESET.

ERROR_32K:
        DJNZ R4, UPDATE_32K     ; IF ERROR OCCURS, REPEAT 3 TIMES.
        .                       ; IN-SYSTEM PROGRAMMING FAIL, USER'S PROCESS TO DEAL WITH IT.
        .
        .
        .
```

12. REVISION HISTORY

VERSION	DATE	PAGE	DESCRIPTION
A3	March, 2002	-	Formerly issued
A4	June, 2004	2	Revise part number in the item of packages
A5	April 20, 2005	34	Add Important Notice
A6	June 7, 2005	3	Add Lead Free (RoHS) parts
A7	October 3, 2006		Remove block diagram
A8	December 4, 2006	3	Remove all Leaded package parts

Important Notice

Winbond products are not designed, intended, authorized or warranted for use as components in systems or equipment intended for surgical implantation, atomic energy control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, combustion control instruments, or for other applications intended to support or sustain life. Further more, Winbond products are not intended for applications wherein failure of Winbond products could result or lead to a situation wherein personal injury, death or severe property or environmental damage could occur.

Winbond customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Winbond for any damages resulting from such improper use or sales.

Please note that all data and specifications are subject to change without notice.
All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.